IN THE CLAIMS

Please cancel claims 2, 4, 13, 18-20, 22 and 27-28.

Please amend the claims as follows.

1. (Currently Amended) An apparatus comprising:

at least one processor;

a memory coupled to the at least one processor;

a database residing in the memory, the database supporting batch updates;

generated code residing in the memory and executed by the at least one processor, the generated code interacting with the database using a plurality of calls to the database; and

a batch mechanism residing in the memory and executed by the at least one processor, the batch mechanism processing the plurality of calls by the generated code and batching a plurality of database updates in the plurality of calls by the generated code to the database, wherein the batch mechanism intercepts at least one call to the database from the generated code, and wherein the at least one call includes an executeUpdate() call.

2. (Cancelled)

3. (Currently Amended) The apparatus of claim [[2]] 1 wherein the at least one call includes a call to prepare a statement.

4. (Cancelled)

5. (Currently Amended) The apparatus of claim [[2]] 1 wherein the at least one call includes a returnPreparedStatement() call.

6. (Original) The apparatus of claim 1 wherein the batch mechanism creates a batch when a call corresponding to a first update is made by the generated code, adds an update to the batch for each subsequent call corresponding to an update that is not a last call, and executes the batch when the generated code makes the last call corresponding to a last update.

7. (Original) The apparatus of claim 1 wherein the batch mechanism batches the plurality of updates without affecting the generated code.

8. (Original) An apparatus comprising:

at least one processor;

a memory coupled to the at least one processor;

a database residing in the memory, the database supporting batch updates;

generated code residing in the memory and executed by the at least one processor, the generated code interacting with the database using a plurality of calls to the database, the plurality of calls including a plurality of calls to a prepareStatement() method, a plurality of calls to an executeUpdate() method, and a plurality of calls to a returnPreparedStatement() method; and

a batch mechanism residing in the memory and executed by the at least one processor, the batch mechanism creating a batch when an executeUpdate() call corresponding to a first update is made by the generated code, adding an update to the batch for each subsequent executeUpdate() call corresponding to an update that is not a last call, and executing the batch when the generated code makes the last executeUpdate() call corresponding to a last update.

9. (Original) The apparatus of claim 8 wherein the batch mechanism closes and caches a prepared statement corresponding to a call when the batch mechanism is done executing the batch.

10. (Original) The apparatus of claim 8 wherein the batch mechanism does not affect the generated code.

11. (Currently Amended) A computer-implemented method for enabling batch processing of database updates without affecting generated code that executes calls to a database that supports batch operations, the method comprising the steps of:

providing a batch mechanism that intercepts a plurality of calls from the generated code to the database, wherein the plurality of calls from the generated code to the database include a plurality of calls to an executeUpdate() method; and

the batch mechanism batching a plurality of database updates in the plurality of calls by the generated code to the database, and executing the batch to the database.

12. (Original) The method of claim 11 wherein the plurality of calls from the generated code intercepted by the batch mechanism include a call to prepare a statement.

13. (Cancelled)

14. (Original) The method of claim 11 wherein the plurality of calls from the generated code intercepted by the batch mechanism include at least one call to a returnPreparedStatement() method.

15. (Original) The method of claim 11 further comprising the steps of:

creating a batch when a call corresponding to a first update is made by the generated code;

adding an update to the batch for each subsequent call corresponding to an update that is not a last call; and

executing the batch when the generated code makes the last call corresponding to a last update.

16. (Original) A computer-implemented method for enabling batch processing of database updates without affecting generated code that executes calls to a database that supports batch operations, the method comprising the steps of:

providing a batch mechanism that intercepts the calls from the generated code to the database, the calls including a plurality of calls to a prepareStatement() method, a plurality of calls to an executeUpdate() method, and at least one call to a returnPreparedStatement() method;

creating a batch when a call corresponding to a first update is made by the generated code;

adding an update to the batch for each subsequent call corresponding to an update that is not a last call; and

executing the batch when the generated code makes the last call corresponding to a last update.

17. (Currently Amended) A computer readable program product comprising:

a batch mechanism that processes a plurality of calls by generated code that interacts with a database using a plurality of calls to the database, wherein the batch mechanism intercepts at least one call to the database from the generated code, and wherein the at least one call includes an executeUpdate() call, the batch mechanism batching a plurality of database updates in the plurality of calls by the generated code to the database; and

~~computer readable signal bearing~~ recordable media bearing the batch mechanism.

18-20 (Cancelled)

21. (Currently Amended) The program product of claim [[20]] 17 wherein the at least one call includes a call to prepare a statement.

22. (Cancelled)

23. (Currently Amended) The program product of claim [[20]] 17 wherein the at least one call includes a returnPreparedStatement() call.

24. (Original) The program product of claim 17 wherein the batch mechanism creates a batch when a call corresponding to a first update is made by the generated code, adds an update to the batch for each subsequent call corresponding to an update that is not a last call, and executes the batch when the generated code makes the last call corresponding to a last update.

25. (Original) The program product of claim 17 wherein the batch mechanism batches the plurality of updates without affecting the generated code.

26. (Currently Amended) A computer readable program product comprising:

a batch mechanism that creates a batch when an executeUpdate() call corresponding to a first update is made by generated code that interacts with a database using a plurality of calls to the database, the plurality of calls including a plurality of calls to a prepareStatement() method, a plurality of calls to an executeUpdate() method, and a plurality of calls to a returnPreparedStatement() method, the batch mechanism adding an update to the batch for each subsequent executeUpdate() call corresponding to an update that is not a last call, and executing the batch when the generated code makes the last executeUpdate() call corresponding to a last update; and

~~computer readable signal bearing~~ recordable media bearing the batch mechanism.

27-28 (Cancelled)

29. (Original) The program product of claim 26 wherein the batch mechanism closes and caches a prepared statement corresponding to a call when the batch mechanism is done executing the batch.

30. (Original) The program product of claim 26 wherein the batch mechanism does not affect the generated code.